OFF-LINE HANDWRITING RECOGNITION USING VARIOUS HYBRID MODELING TECHNIQUES AND CHARACTER N-GRAMS

Brakensiek, A.; Rottland, J.; Kosmala, A.; Rigoll, G.

Published in:
EPRINTS-BOOK-TITLE

OFF-LINE HANDWRITING RECOGNITION USING VARIOUS HYBRID MODELING TECHNIQUES AND CHARACTER N-GRAMS

A. Brakensiek, J. Rottland, A. Kosmala, G. Rigoll
Dept. of Computer Science, Faculty of Electrical Engineering
Gerhard-Mercator-University Duisburg
47057 Duisburg, Germany
E-mail: {anja, rottland, kosmala, rigoll}@fb9-ti.uni-duisburg.de

In this paper a system for off-line cursive handwriting recognition is described. The system is based on Hidden Markov Models (HMMs) using discrete and hybrid modeling techniques. Here, we focus on two aspects of the recognition system. First, we present different hybrid modeling techniques, whereas one depends on an information theory-based neural network (MMI-criterion) used as a vector quantizer and the other uses a neural net for estimating the a posteriori probabilities to replace the codebook of a tied-mixture HMM system. This is the first paper where we present this novel approach -called tied posteriors- for handwriting recognition. Second, we demonstrate the usage of a language model, that consists of character n-grams, as an alternative to the recognition with a large dictionary of German words. Our resulting system for character recognition yields significantly better recognition results using an unlimited vocabulary.

1 Introduction

During the last years, Hidden Markov Models (HMMs, see[1]) have been used as one of the most popular paradigms for on- and off-line handwriting recognition (for example[2,3,4,5]) and for segmentation-free recognition of degraded machine-printed documents (as in[6,7]).

Crucial components of a cursive handwriting recognition system[8] are efficient preprocessing operations (as in[9]), a robust feature extraction (compare also[10]) -particularly with regard to a writer-independent system- and the modeling approaches as well as the usage of contextual knowledge.

The emphasis in this paper is on the comparison of different modeling techniques for a writer-dependent off-line handwriting recognition system using a large (30k) vocabulary of German words. We compare discrete HMMs with two different hybrid approaches, where the HMM is augmented by a neural network, that can be used either as an approximator of the probability density function for (semi-) continuous HMMs (see[11,12]) or as a neural vector quantizer for discrete HMMs[4,13].

Another focus is on the usage of contextual knowledge for language models on character level[6] (backoff n-grams) instead of a given closed dictionary. So

In: L.R.B. Schomaker and L.G. Vuurpijl (Eds.), Proceedings of the Seventh International Workshop on Frontiers in Handwriting Recognition, September 11-13 2000, Amsterdam, ISBN 90-76942-01-3, Nijmegen: International Unipen Foundation, pp 343-352

an improvement of character recognition results with unlimited vocabulary can be achieved.

In the following sections our basic off-line recognition system (Section 2) including the description of the database and feature extraction methods, the different HMM modeling techniques (Section 3) and the use of language models (Section 4) is presented. The recognition experiments and results are given in Section 5. Finally, Section 6 summarizes the presented work and gives an outlook on additional future tasks.

2 Basic Handwriting Recognition System

Our recognition system consists of about 80 different linear HMMs (compare [13]), one for each character (upper- and lower-case letters, numbers and special characters or punctuation marks like ',:(?'). Mostly, there are used eight states per HMM for numbers and letters and about four or fewer states for some special characters depending on their width. To train the HMMs we use the Baum-Welch algorithm, for recognition the Viterbi algorithm is used.

The presented recognition results refer either to a word error rate depending on a 30k lexicon or to a character error rate, which depends on substitutions, insertions and deletions of characters. A reason for using the character- instead of the word-recognition rate is the use of an unknown and unlimited vocabulary in real-world documents (where a greater vocabulary than 30k is usual), such as comments on special topics or actual news.

2.1 Database

The database consists of cursive script samples of four different writers (ABR, ANK, JMR, VDM), all writing a training set of some sentences (about 2000 words in lower and upper case) and a test set of nearly 200 single words (about 1200 characters) on a digitizing surface. This on-line input is transformed into a pixel-bitmap, so that no further preprocessing like skeletonizing or noise reduction is necessary in contrast to scanned images. The dynamic features of this on-line input information are not used. Examples of the test set are shown in Fig.1.

2.2 Feature extraction

In off-line handwriting recognition the following features are derived from the database (see also [13]):

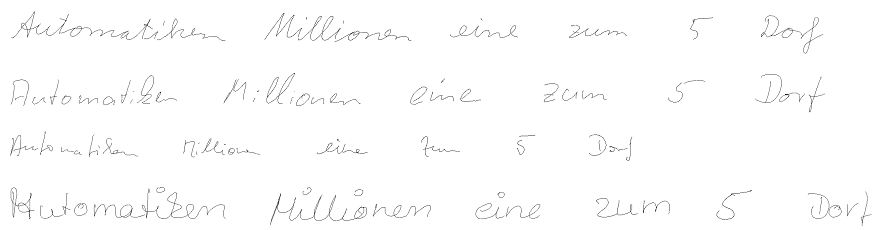

Figure 1: some examples of the test set (writers: ABR, ANK, JMR, VDM)

- DCT-coefficients (Discrete Cosine Transform) of a small bitmap slid along the horizontal direction (8-dimensional feature vector), whereas the surrounding defines the current height and position of this area
- some complementary features such as height over baseline, the thickness at the current horizontal position and number of black-white transitions (3-dimensional vector)

Necessary for feature extraction is the estimation of the baseline, which is done by an approximation of a horizontal line to the local minima of the word, respectively sentence. Normalization of the input-data implies only the correction of the skew, whereas slant correction and height normalization is not absolutely necessary in a writer-dependent system, taking into account that one attribute of HMMs is the capability of generalization.

Fig.2 illustrates the most important aspects of the feature extraction. Attributes of the bitmap-feature are independent of the baseline, whereas the additional features depend on baseline and character height. A more detailed description can be found in [13].

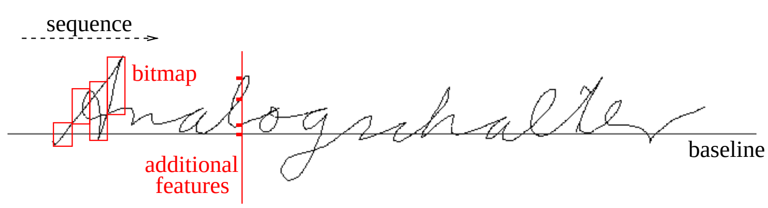

Figure 2: feature extraction for off-line recognition

These 11-dimensional feature-vectors are used for training and testing the off-line system. Using discrete HMMs this vector $\underline{x}$ is quantized by two separate codebooks (VQs) of size 200 and 100 (corresponding to the two kinds of features, which are extracted). The sizes of the neural VQs for the first kind of hybrid HMMs (based on a discrete structure) and the corresponding k-means VQs for discrete HMMs are equal. Using the second variation of hybrid tied-mixture HMMs (see Section 3), the entire feature-vector $\underline{x}$ is processed in one single stream.

3 HMM Modeling Techniques

In general, there are three different modeling techniques used for HMMs: continuous, discrete or hybrid (see [1,4,12]). In this paper recognition experiments using a discrete and two different hybrid approaches, which consist of a discrete and a semi-continuous structure, are compared. The semi-continuous hybrid approach was recently developed and tested by our research group [12] and is presented here for the first time for handwriting recognition. In that way, this paper can be considered as a continuation of the work presented in [13], where the major advancement here is the comparison of the new tied-posterior approach to the techniques investigated already in [13].

Using discrete HMMs the feature vector $\underline{x}$ is first processed by a vector quantizer (VQ), which leads to the generation of a VQ label y_n according to:

$$\underline{x} \overset{VQ}{\rightarrow} y_n \tag{1}$$

By using this k-means VQ the probability of vector $\underline{x}$ in state s is approximated by the probability of the label y_n as:

$$p(\underline{x}|s) = p(y_n|s) \tag{2}$$

Using the hybrid approach based on discrete HMMs as described in [4], a winner-takes-all neural network replaces the k-means VQ. This neural VQ is trained according to the MMI (Maximum Mutual Information) criterion:

$$I(Y,W) = H(Y) - H(Y|W) = H(W) - H(W|Y) \tag{3}$$

where Y is the sequence of firing neurons, W the string of classes (characters) corresponding to the training vector sequence X and H the entropy (in the following, this approach is called hybrid-MMI). Due to the structure of these hybrid HMMs, the recognition is as fast as using discrete HMMs.

The third alternative (details are described in [12]), which is presented here, is the use of a further hybrid modeling technique, which is based on tied-mixture technology. Here, the HMM is augmented by a neural network, that is used for estimating the a posteriori probabilities to replace the codebook of a tied-mixture system (called hybrid-TP = tied-posterior). As network architecture, we use a multi-layer-perceptron (MLP) with one hidden layer.

The emission probabilities b_i for each state s_i of a tied-posterior system can be computed as follows:

$$b_i(\underline{x}) = p(\underline{x}|s_i) = \sum_{j=1}^{J} c_{ij} \cdot p(\underline{x}|j) = \sum_{j=1}^{J} c_{ij} \cdot \frac{p(j|\underline{x})}{p(j)} \tag{4}$$

In this approach the conditional probabilities $p(\underline{x}|j)$ (Gaussian pdfs) of a usual tied-mixture HMM are replaced by the a posterior probabilities $p(j|\underline{x})$, which are estimated by the neural net, and the a priori class probabilities $p(j)$ using Bayes rule (J is the number of tied mixtures, respectively the size of the codebook). The weighting factors c_{ij} have to be estimated by the Baum-Welsh algorithm, as in the usual tied-mixture technique. If the weighting factors c_{ij} are chosen in such a way, that only one of the network outputs is active for each HMM, according to

$$c_{ij} = 1 \quad if \quad i = j \quad and \quad c_{ij} = 0 \quad else \tag{5}$$

this tied-posterior system is transformed into a standard hybrid approach considered in [11].

We believe that this new hybrid approach offers several potential advantages. Among them are the following:

- As shown in [11] MLPs are excellent posterior probability estimators and can lead to high recognition rates for systems with a small number of parameters. Our approach is an useful extension to HMMs with many states, as they are usually required for handwriting.

- Compared to a standard system based on Gaussian tied-mixtures, our tied-posterior system has the advantage that it can handle multiple frame input very well and that the codebook size can be kept small while using feature vectors combined out of different features (which usually require large codebooks or multiple codebooks for standard HMMs). Thus we can generate a system equivalent to a tied-mixture system but with much less parameters and multiple frame input.

4 Language Modeling

As an alternative to the 30k dictionary of German words, our handwriting recognition system can be used without lexicon on character level. To improve recognition performance when no dictionary is available, we use language models (backoff n-grams), which are well known in speech recognition, on the character level. This model influences the transition probabilities between the trained character HMMs. Using Bayes rule the solution for our character recognition problem can be described as follows:

$$W^* = \operatorname*{argmax}_W \ p(W|X) = \frac{\operatorname*{argmax}_W p(W) \ p(X|W)}{p(X)} = \operatorname*{argmax}_W \ p(W) \ p(X|W) \tag{6}$$

with $p(X|W)$ presenting the feature model (describing the stochastic relation between the features and the HMM states, as outlined in Sec.2.2 and 3) and $p(W)$ the grammar or language model (see also [6,14]). The probability of the features $p(X)$ is nonrelevant, because it is independent of the classes W.

This language model is described by a backoff n-gram of characters (not words) with n=3 or 5. It takes into account that, for example, the character sequence 'qu' is much more probable than 'qo'. The formula for estimating a backoff bigram is the following

$$P(w_2|w_1) = \begin{cases} N(w_1, w_2) \cdot d)/N(w_1) & : \ N(w_1, w_2) > t \\ p(w_2) \cdot b(w_1) & : \ else \end{cases} \tag{7}$$

with $N(w_1, w_2)$ the number of times character w_2 follows w_1. The discounting coefficient d and the backoff factor $b(w_1)$ are necessary to correct the probabilities for observed and unseen events (see [15]).

Examples for applications of language models are the use of bi- or trigrams of characters (compare also [6]) for word recognition, as described in this paper or n-grams of words in order to enhance sentence-recognition or document classification.

We generate the n-gram model (including also special characters) by using the statistical character-sequences of about four millions of words from German documents (several HTML-pages), which leads to 195877 (19505) different 5-gram (3-gram) sequences.

It should be noticed, that this database, that is different from the handwritten dataset, is quite easy to create, because only the ASCII-text (and no image) is necessary. It is also worthwhile to mention here that incorporating n-grams directly into the decoding process is not a trivial task at all (especially if $n \geq 3$) and requires the usage of special decoders or decoding procedures. This is one of the major reasons why most recognition systems are not capable of using large n-grams.

The language model is trained with the CMU toolkit (see [15]) and a more detailed description of the decoding procedure using this kind of language model can be found in [14].

5 Recognition Experiments

We tested the influence of two different factors on the recognition performance: first, the modeling technique, which compares discrete with both kinds of hybrid HMMs, presented in Sec.3, and second, the inclusion of language models consisting of character n-grams (see Sec.4) as an alternative to a large dictionary.

Table 1 presents word recognition results using a 30k dictionary, which enable a direct comparison of different modeling techniques for four writers. A separate recognition system for each writer (writer-dependent mode) is used for the following tests. All experimental results are obtained using multiple frame input of the feature vectors, which takes the context of the current features into account.

In average the relative error can be reduced by about nearly 20% using hybrid-MMI HMMs (recognition rate of 89.2%) instead of discrete ones (comparing the first two rows of Tab.1).

The same effect can be observed comparing the results using our system in writer-independent mode (one system for all writers). Here the accuracy is 79.9% using discrete HMMs respectively 83.8% with the hybrid-MMI approach. Certainly, this is lower than the obtained average, taking into account, that no further preprocessing of the database has been made to equalize the different writing styles.

Most errors are confusions concerning the word ending, which is often written unclear, or permutations of single letters only, which are quite similar (e.g. '5' $\leftrightarrow$ 'S' or 'kam' $\leftrightarrow$ 'kann').

Experiments with hybrid-TP HMMs yield lower recognition rates (86.3% in average compared to 89.2% for hybrid-MMI HMMs), which can be explained by the kind of the feature vectors and the relative small training set. The recognition rate is comparable with those of discrete HMMs. For estimating the posterior probabilities we use a MLP with one hidden layer (300 units) and an input layer of size $(11 \cdot \#frame)$, the output-size is determined by the number of classes. Here, the number of adjacent frames is set to $\#frame = 15$ in comparison to three frames, which are optimal for the discrete modeling structure (referring to the results presented in Tab.1). Crucial for this kind of modeling is the number of frames and states. So, the recognition results decreases significantly using only one state, as is usual for standard hybrid approaches like [11], which is a special case of our tied-posterior modeling technique.

Since the best word recognition rates could be obtained using hybrid-MMI HMMs, the next experiments use this method for a comparison of different character n-grams. In Table 2 character recognition results (counting substi-

Table 1: Off-line handwriting recognition results (word accuracy in %) using different modeling techniques (30k dictionary)

method	ABR	ANK	JMR	VDM	average
DISCRETE	98.4	92.3	78.3	77.7	86.7
HYB-MMI	98.9	93.4	80.4	84.2	89.2
HYB-TP	96.2	92.9	78.3	77.7	86.3

tutions, insertions and deletions of characters) using the hybrid-MMI modeling technique are shown.

As expected, the recognition accuracy decreases, when using no dictionary. The character recognition rate of 75.4% implies a word recognition rate of only 36% (compared with 89.2% with given vocabulary).

Using n-grams for character recognition, the accuracy increases significantly, so that in average a character recognition rate of about 83.1% (using trigrams) can be obtained. This means a relative error reduction of about 30% compared to the results without language model. A n-gram of higher context depth (5-gram) provides a further error reduction. Here, the character accuracy is 87.9% in average. This means a word recognition rate of about 65% (compare Tab.1).

Most errors are confusions between the characters 'm', 'n' and 'r' or 'e' and 'a' or 'i' and 'ü'. Insertions affects mostly the characters 'r', 'n' and 'c'. One kind of errors occurs, if characters are quite similar, an other kind occurs, if it is possible to split a character in two valid characters ('m' $\rightarrow$ 'rn') or vice versa.

Table 2: Off-line handwriting recognition results (character accuracy in %) using hybrid HMMs and different n-grams (without dictionary)

method	ABR	ANK	JMR	VDM	average
HYB-MMI: 1-gram	90.7	79.2	65.2	66.6	75.4
HYB-MMI: 3-gram	94.2	87.7	75.1	75.2	83.1
HYB-MMI: 5-gram	96.0	91.9	81.4	82.4	87.9

The shown experiments with character n-grams lead to the assumption, that a higher context depth will provide significantly smaller error rates. So character n-grams that span over word boundaries might become useful.

6 Summary and Outlook

We presented in this paper a HMM based off-line handwriting recognition system in writer-dependent mode. The focus is on different HMM modeling techniques and the usage of language models compared to dictionaries.

The above experiments of this segmentation-free approach show the better performance of a hybrid modeling technique for HMMs, which depends on a neural vector quantizer (hybrid-MMI), compared to discrete and hybrid HMMs, based on a tied-mixture structure (hybrid-TP), which may be caused by a relative small data set. The relative error rate for word recognition can be reduced by about 20% using a large vocabulary of German words.

Additionally, we describe the influence on character recognition accuracy (without dictionary) obtained by the use of a language model based on character n-grams. Here, the relative error rate decreases by about 50% using 5-grams instead of using no language model. As expected, the word accuracy decreases using no dictionary by about 26%.

Future work has to imply a writer-independent recognition system with focus on a more expendable preprocessing and feature extraction (see also [10]) and experiments on a public, larger database (for example like [16]), which consists of whole sentences to take the word grammar into account.

References

1. L.R. Rabiner and B.H. Juang. An Introduction to Hidden Markov Models. *IEEE ASSP Magazine*, pages 4–16, 1986.
2. R. Seiler, M. Schenkel, and F. Eggiman. Off-Line Cursive Handwriting Recognition Compared with On-Line Recognition. In *Proc. Int. Conference on Pattern Recognition (ICPR)*, pages 505–509, Vienna, 1996.
3. R.-D. Bippus and V. Maergner. Cursive Script Recognition Using Inhomogeneous P2DHMM and Hierarchical Search Space Reduction. In *5th International Conference on Document Analysis and Recognition (ICDAR)*, pages 773–776, Bangalore, India, 1999.
4. G. Rigoll, A. Kosmala, and D. Willett. A New Hybrid Approach to Large Vocabulary Cursive Handwriting Recognition. In *International Conference on Pattern Recognition (ICPR)*, pages 1512–1514, Brisbane, August 1998.
5. J. Franke, J.M. Gloger, A. Kaltenmeier, and E. Mandler. A Comparison of Gaussion Distribution and Polynomial Classifiers in a Hidden Markov Model Based System for the Recognition of Cursive Script. In *Proc. Int. Conference on Document Analysis and Recognition (ICDAR)*, pages 515–518, Ulm, Germany, August 1997.

6. I. Bazzi, R. Schwartz, and J. Makhoul. An Omnifont Open-Vocabulary OCR System for English and Arabic. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(6):495–504, June 1999.
7. A. Brakensiek, D. Willett, and G. Rigoll. Improved Degraded Document Recognition with Hybrid Mo deling Techniques and Character N-Grams. In *15th Int. Conference on Pattern Recognition (ICPR)*, Barcelona, September 2000.
8. G. Kim, V. Govindaraju, and S.N. Srihari. Architecture for Handwritten Text Recognition Systems. In *6th International Workshop on Frontiers in Handwriting Recognition (IWFHR)*, pages 113–122, Taejon, Korea, 1998.
9. M. Schüßler and H. Niemann. A HMM-based System for Recognition od Handwritten Adress Words. In *6th International Workshop on Frontiers in Handwriting Recognition (IWFHR)*, pages 505–514, Taejon, Korea, 1998.
10. L. Schomaker and E. Segers. Finding features used in the human reading of cursive handwriting. *Int. Journal on Document Analysis and Recognition (IJDAR)*, 2:13–18, 1999.
11. H. Bourlard and N. Morgan. Connectionist Speech Recognition - A Hybrid Approach. *Kluwer Academic Press*, 1994.
12. J. Rottland and G. Rigoll. Tied Posteriors: An Approach for Effective Introduction of Context Dependency in Hybrid NN/HMM LVCSR. In *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Istanbul, June 2000.
13. A. Brakensiek, A. Kosmala, D. Willett, W. Wang, and G. Rigoll. Performance Evaluation of a New Hybrid Modeling Technique for Handwriting Recognition Using Identical On-Line and Off-Line Data. In *5th International Conference on Document Analysis and Recognition (ICDAR)*, pages 446–449, Bangalore, India, September 1999.
14. D. Willett, C. Neukirchen, and G. Rigoll. DUCODER-The Duisburg University LVSCR Stackdecoder. In *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Istanbul, June 2000.
15. P. Clarkson and R. Rosenfeld. Statistical Language Modeling Using the CMU-Cambridge Toolkit. In *Proc. Eurospeech '97*, pages 2707–2710, Rhodes, Greece, September 1997.
16. U.-V. Marti and H. Bunke. A full English sentence database for off-line handwriting recognition. In *5th International Conference on Document Analysis and Recognition (ICDAR)*, pages 705–708, Bangalore, India, 1999.